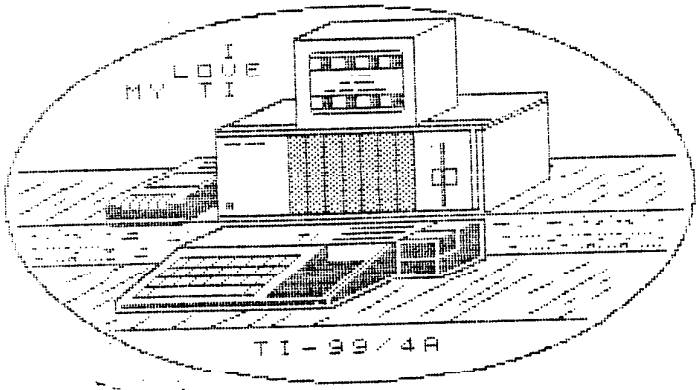


Topics

DISCLAIMER
Any opinions or statements in this newsletter, whether in editorial form, articles, reviews, or material copied from other newsletters, are those of the individual authors only, and do not represent others in the LA 99er User Group or the group as a whole.

LA 99er COMPUTER GROUP

VOL.7 NO.5 LOS ANGELES, CA MAY, 1988

Newsletter

T n T

SOME OF THIS AND SOME OF THAT

Oops! Our good friend and talented programmer J. Peter Hoddie is the Editor of the BCS Newsletter. Our good friend Mike Wright is their Commentator. I knew that, just blew it.

George Steffen report, as of last night May 16 he was still in the Bronx VA. His daughter Susan called with the information that he was finally strong enough to stand up. Hopefully home soon. Just awful not to be able to call him. How about some of our New York City friends making a hospital visit? Susan also asked where to buy game modules for her son Stephan as a birthday gift. Living in Lake Tahoe, it is not an easy task. If anyone has some extra, it would be a nice gesture for Georges' grandson. Stephan Lehrer, PO Box 11073, Zephyr Cove, Nevada 89448.

DALLAS FAIRE

Thanks to the hospitality of Annie and Richard Fleetwood, Barry Traver and I were able to attend the first Dallas Faire. Air fare and hotel were more than I could consider, so Annie and Richard opened their home to us. The efforts of the Dallas people certainly showed the Texas spirit. The set-up was very good. Surprisingly we found systems in place and all ready to go. The strongest supporters of User Group Faires were there, the Mc Deavitts of RAVE fame, Gene Chandler and Richard Pacquette of Great Lakes Software, Bud Mills of Horizon Ram Disk, Bob Boone of Ottawa, Jim Horn of Compuserve, Scott Darling of GEnie. Hope I did not leave anyone out, sorry if I did. Credit MUST be given to the time, effort and monies spent by these fine folk. The best way to thank them is support and buy their products. Thanks to PLUG and Richard Fleetwood for their fine show. Perhaps next year the Dallas User Group can be the host. Why not alternate yearly? And in case you were not there I now own a Mike Stanfil Graphic T-Shirt personally autographed by the artist himself (while I was wearing it!). Right over the heart. Thanks Mike I enjoyed every stroke.

SOLITAIRE

During TICOFF Larry Hughes of Quality Soft presented me with a copy of his QS Solitaire. He knew how much I enjoyed Walt Howes Chainlink Solitaire. Well I can heartily recommend QS-Solitaire. It is a fine and challenging game, both of them actually. Just wish I could relocate it to my Ram Disk for rapid access.

Anyone wonder why Topics is not mentioned in Computer Shopper? Bet you won't get an answer. Et tu Brutus?

FEST-WEST'89

Our good friend Woody called recently to advise of their progress in locating a suitable site. We are so happy that they have taken the ball and are running with it! San Diego is a great location for a Fest. A good Hotel with meeting area and banquet space was great in both Las Vegas and Dallas. The set-up is so much easier. Woody and friends, if any of us can help at all please ask.

WHITE SOMBRERO

Gail Fair, our incredibly loyal, reliable quiet friend has long deserved this accolade. Gail has never missed a club meeting, board meeting, swap meet. He has ALWAYS been there for everyone of us. Keeper of the Marketplace, schlepper extraordinaire. There is NO way to adequately say THANK YOU to Gail, he is incredible.

NOTE NEW ADDRESS PLEASE
L.A. 99ers COMPUTER GROUP P.O. BOX 67A79 LOS ANGELES, CA. 90067

GRAM KRACKER RAFFLE

Steve Chalcraft, a very participatory member of our club, and owner of one of our two BBS was the lucky winner. Really deserved it too.

Next Raffle prize is a MYARC 512 Ram Disk. Buy your tickets now!

NEWSLETTER EXCHANGE

Thanks to postal rate increase, beginning this month we will be mailing our exchange newsletters two at a time every other month. Rather do this than increase membership. If you belong to another group also, please pass this information along to them. thanks.

COMMENT

We do not censor any material printed, we clearly do not always agree with content, but strongly support free press.

RAFFLE RAFFLE RAFFLE RAFFLE

The Club Raffle for the months of May and June (and possibly July depending on how tickets go!) will be for a Myarc Floppy Disk Controller, with 80 track eprom. This FDC has to be seen in action to be believed. Of the three FDCs available for the TI market, this is by far the fastest in reading or writing to disk. I won't go into the technical details, but believe me it is true. The original Myarc FDC was for DSDD drives - in other words it could format disks up to 1440 sectors. The 80 track model for the raffle can use the usual 48 tpi drives in this way. However it is also capable of using 96tpi drives and then formatting disks to 2880 sectors. This is 720Kb on one disk, or 3/4 of a megabyte. Imagine that! It comes complete with the Myarc Level III disk manager on disk, which is also about the best that I know (it works ONLY with the Myarc FDC).

In order to induce you to buy as many tickets as possible - this is after all a fund raising activity for the Club - the price of tickets has been modified so that the more you buy the cheaper each one becomes. Remember that the more tickets you buy the greater chance you have of winning.

In order to give our national and international members a chance to join in, each raffle goes for two months, so the drawing for the disk controller will be at the June or July meeting. If your check misses this raffle, your entry willbe applied to the next one.

Please send your name, address, and money to the club address. You do not need this form if you don't like to cut up your newsletter. Any piece of paper will do.

NAME _____

ADDRESS _____

CITY _____

STATE, ZIP _____

1	TICKET	$2.00	_____
3	TICKETS	$5.00	_____
7	TICKETS	$10.00	_____
11	TICKETS	$15.00	_____
15	TICKETS	$20.00	_____
20	TICKETS	$25.00	_____
25	TICKETS	$30.00	_____
30	TICKETS	$35.00	_____
36	TICKETS	$40.00	_____
42	TICKETS	$45.00	_____
50	TICKETS	$50.00	_____

DISK CONTROLLERS FROM TI TO MYARC

Copyright by Jerry Coffey January 1987

[Ed. Note: I first saw this article over one year ago, and immediately filed it as by far the best and most cogent one on this piece of hardware that I have ever seen. We asked Jerry to send us the original article, so that we could format it to our own standards rather than risk a poor copy from another newsletter. Our apologies to whichever newsletter it was originally written for - I can't remember right now which one it was!]

The views expressed in this article reflect the author's personal experience with TI, Corcomp, and Myarc disk controllers. Technical data has been verified wherever possible, but is not publicly documented in some instances. Please bring any errors to the attention of the author.

The disk capacity of the TI99 has increased in just a few years from less than 80K (a single one-sided 35 track drive) to almost 2.9 megabytes (four double-sided, double-density, 80 track drives). The early standalone was replaced by the PEBox system which would support three double-sided 40 track drives (540K). Corcomp introduced their four drive double-density system (1440K), followed by Myarc's similar system with two double-density formats (1280K and 1440K). Then in 1986, Myarc offered its 80 track upgrade which doubled capacity again. Even as capacity was increasing rapidly, the TI and Corcomp controllers differed only modestly in I/O speed. When MYARC introduced its fast DSDD controller, few reviewers did justice to its speed advantage. Early comparisons were done at the standard TI or Corcomp interlace, but the big speed gains required taking advantage of the much tighter sector interlace possible with the high-speed MYARC card. To understand how this works we need to take a look at the way a disk drive performs.

Disk Drive Fundamentals

A floppy disk drive writes information in concentric rings called "tracks" on a thin plastic disk coated with a film of magnetic particles. Each track in turn is divided into blocks of information called sectors. A blank disk has one (or more) index holes used to synchronize the process of writing to and reading from the disk. The type with many holes are called "hard sectored" since each sector has its position fixed by an index hole. The type of disks used by most computers have only one hole and are called "soft sectored". In this system the computer must write magnetic signposts on the disk to mark out each sector in a process called "formatting" or "initializing" a disk. These signposts take up a subtantial fraction of the space on a track since they include not only sector numbers but buffers (filler bytes) that allow the computer to get into synchronization to read or write sectors of data and to prevent the sector identifier from being overwritten by a drive operating at a slightly different speed from the drive that formatted the disk.

The typical 5.25 inch disk drive has a "stepper motor" capable of moving the drive's read/write head(s) in or out along a radius of the disk in steps of 1/40 of an inch (thus the terminology "48 tpi" = 48 tracks per inch). Since the inner tracks have a smaller circumference, they crowd the bits of information together. Magnetic coatings on a floppy disk are rated by their capacity in bits per inch at standard magnetic flux for the write head. This figure is usually over 5000 bpi for modern floppies, but was somewhat lower a few years ago. The circumference of the inner track of a 40 or 80 track disk is about 10 inches -- which allows about 6250 bytes to be written on the track without exceeding 5000 bpi. For comparison, the Corcomp double density format requires over 6400 bytes per track. Media limitations were the reason that some early 5.25 disk drives only used the outer 35 tracks. The 16 sector (by 256 bytes/sector) format recommended by most drive makers requires only 6250 bytes per track and includes several hundred additional "buffer" bytes to compensate for differences in drive timing.

Timing is EVERYTHING

With soft-sectored disks, the integrity of the read/write processes require critical timing. The disk rotates at 300 rpm within a small margin. This means there are about 250 thousand magnetic pulses (bits) passing beneath the head each second. In single density format, the majority of these pulses are timing or filler bits -- in double density, many of the timing bits are suppressed in order to double the rate of data bits. In a typical sector read the drive must bring the disk up to speed, recognize the index hole, step out to track zero (to get its bearings), determine single or double density, verify its position, step in to the target track, verify the track number (written in the format operation), detect the sector identifier as it flies past, then immediately read the 256 data bytes into memory. Five of these operations require accurate reading of the magnetic pulses whizzing by at over 250K bits per second.

If you do some quick arithmetic (256 bytes/sector = 2048 bits/sector into 250k bits/second)... hmmm... Why can't the drive read a 125 sector file in one second? Well first many of those bits are not data bits, they are overhead to keep things synchronized and allow for timing variation between drives. Second, some time is used moving the head from one track to the next when more than one track must be read. Third, 250K is the instantaneous read rate and the computer must take time to do other

things like move the last sector out of its buffer to make room for the next one. In the standard TI protocol for reading a disk, the data is moved into VDP ram (so the drive could be used without the memory expansion) before it goes to the expansion memory. All this thrashing eats great chunks of the time available for reading data. By the time one sector is safely tucked away in the 32K card, several sectors have already passed by the drive's read head. If the sectors were written consecutively on the disk, we would have to wait a full revolution (0.2 seconds) before the next sector would pass under the head. To avoid this inefficiency, the consecutively numbered sectors are spaced out around the disk so that they are separated by just enough time to take care of other business. The actual pattern in which the sectors are scattered is called the "interlace". The idea of the interlace is to spread the sectors out to match the timing needs of the hardware -- both the time needed to stash each sector and the time needed to step from one track to the next and get the the head settled down for some serious (250K bps) reading.

Interlace and Head Step Times

Life was simple with the TI disk controller. Both the interlace and the head step time were locked into the controller's PROM (that's the programmable chip that contains the control programs for the card). The head step time is the built-in delay between step signals to allow the stepper motor to move the head one "click" in or out. The TI settings are very conservative (read "slow") to allow for slow drives. The step time is 20ms -- if you step from track zero to track 39, it takes 20x39=780ms, almost four revolutions of the drive. The TI interlace lays the sectors down on a track in the order 0753186420. This allows all sectors to be read in four revolutions of the disk though the slow head step lets another revolution go by between tracks. Thus the maximum read rate is about 9 sectors per five revolutions (= one second) or 2304 bytes per second.

When Corcomp designed its double density disk controller, allowances were made for the increased speed of later drives by permitting the step rate to be set with DIP switches for each drive. The step rates available are 30, 20, 12, and 6ms (the faster values quoted in the CC manual are referenced to the wrong clock speed). They also provided a choice of interlace options, though only a couple of them are practical. The default interlaces are labeled "7" for single density and "10" for double density. The single density interlace is the same as TI's, but with a faster step setting the head be can moved without losing a revolution and thus reads 29% faster than the TI controller. The double density interlace allows 18 sectors to be read in five revolutions, but it doesn't leave enough margin to stash the last sector and step the head in time to catch the zero sector of the next track (that's why the sector number "hangs" for 0.2 seconds each 18 sectors while verifying a formatted disk -- you are seeing the extra revolution needed to acquire the first sector of the next track). Thus the maximum read rate is 18/1.2 or 15 sectors per second, about 67% faster than the TI controller. Users of the CC controller have probably noticed that it loads its own MANAGER program faster than this. In this case a special loader bypasses VDP and loads directly to CPU RAM -- this faster handling of the data allows the stepper motor to be activated sooner and saves one revolution per track (so the 98 sector file can be read in about 5.5 seconds). This provided a foretaste of the speed that MYARC would achieve with its double density controller.

The MYARC controller bypasses VDP RAM to load directly to CPU RAM. This technique coupled with a buffer RAM chip on the controller card provided a quantum jump in disk I/O speed. The MYARC card reads the TI single density interlace at 11.25 sectors/second (the same as Corcomp) and reads the CC 18 sector/track interlace at 18 sectors/second (the same speed Corcomp reads its MANAGER program), but this is only the beginning. Since the hardware empties its sector buffer faster, consecutive sectors can be placed closer together allowing a track to be read in fewer revolutions, i.e., it supports a faster interlace. With fast drives, the 9 sector/track single density format can be read at interlace "2". (NOTE: In the MYARC terminology, the interlace number represents the number of disk revolutions required to read a track.) This works out to 22.5 sectors/second compared to 9 for the TI and 11.25 for the CC controller. The MYARC 16 sector format can be read at interlace "3", 26.67 sectors/second -- 3 times as fast as the TI controller and almost twice as fast as Corcomp double density. The Corcomp 18 sector format can be read at interlace "3" or "4", but the data rate is the same in either case, 22.5 sectors/second. Interlace "4" is smooth but requires a very quick head step, interlace "3" reads the track in 3 revolutions but forces an extra revolution for the step from track to track because sectors 17 and 0 are adjacent on the disk. Though both interlaces have the same data rate, interlace "3" is safer if you are uncertain about the speed of your stepper motor.

In order to read and write both double density formats, the MYARC system must insert an additional step in some I/O operations -- sector zero must be read to determine whether a double density disk has 16 or 18 sectors per track. This datum is needed to convert the the logical sector numbers used by the TI operating system into track and sector-within-track addresses for the floppy disk controller chip. The TI and Corcomp controllers do not need this step because they do not use the full potential of the TI disk I/O protocol. Once this step, accessing sector zero, is added to the various disk operations, it opens the system up for using more than two formats -- including 80 track formats.

Beyond Double Density

A two format system can be managed using only the floppy disk controller's inherent ability to sense single

and double density recording patterns. To get beyond this limitation, the additional data stored in sector zero must be read, stored, and used to modify the special binary commands sent to the FDC (floppy disk controller) chip. Fortunately the TI99/4A system design already provides for such innovations through the Device Service Routine concept and standard "GPL" calls. The system doesn't care what hardware is attached as long as it plays by the rules -- an interface program stored in a memory chip (PROM) on the peripheral device does the trick. This program handles calls for I/O operations from other programs such as TI Writer or the Basic Interpreters. Another set of rules controls the way disk and file information are saved on a disk. Disk parameters are stored in sector 0, while sector 1 must have a two byte "pointer" (a hexadecimal sector address) for each block (one sector) containing the bookkeeping data for a file. It is these blocks that are scanned in order to display the disk directory

Since the Myarc controller must read sector zero to determine the number of sectors per track, the other parameters in that sector are available to control other variables such as number of tracks. But there were other limitations to overcome. The number of files on a disk is limited by the space available for pointers. 256 bytes at 2 bytes per pointer would give 128 files -- except the pointer list must end with a null word (>0000) so directory routines know where to stop -- so we get 127 files per disk. The pointer itself can address sector numbers as high as 65535, so this is no problem. The real limitation is the bit map in sector 0. It begins at byte 56 leaving only 200 bytes or 1600 bits available to map the disk. Since a bit must be turned on for each sector used, the 1440 sector DSDD 40 track disk is already near the limit. The answer devised for the 80 track DSDD system is to map two consecutive sectors with each bit. It wastes some space but no more than systems that use a standard 512 byte sector.

Making the Quad System Work

So now lets say we have new code in the disk controller EPROM (an "erasable" version of the PROM chip used by TI) that does all the proper tricks with the bit map and has the FDC commands to control the new 80 track drives we have added to the system. We still have to tell the controller which drives are 80 track and find a disk manager program that can use the new commands. The selection problem can be taken care of using the DIP switches on the card (but in the process you lose their original function -- setting step speed). Since the Eprom responds to standard GPL calls, most functions can be handled by the TI Disk Manager 2 cartridge. The exception is the disk formatting process -- the formatting works OK, but the initial data written into sector zero is for the standard bit map. (This can be fixed by changing byte 56 from >03 to >01 with a sector editor.) Read/write operations from XB or TI Writer work fine since they use the GPL protocols. Myarc has an excellent disk manager program that works beautifully with 40 track drives, but it has suffered from a number of subtle bugs in 80 track mode. This program, like many others designed for high speed I/O, uses assembly language code to handle the FDC -- bypassing some of the routines in the EPROM. Differences in bit map handling, even slight differences in execution times can affect the performance of 80 track drives. The code in the 80 track EPROM has had a lot of attention to proper timing -- the price you pay for higher performance.

Fine Tuning the Myarc Disk System

Before you start using the Myarc system routinely, there are some experiments that can get maximum performance from your drives. Use the Myarc disk manager to try different interlace settings -- first with your 40 track drives, then with the 80 track drives. Watch for hesitations as each formatted disk is verified, then use the Test option to read the sectors you have layed down. Look and listen for "retries" -- when the sector number pauses with a head seek noise. Use the best disks you have and note the combinations that test smoothly. With fast drives in good condition, you should be able to run 9 sector (single density) format at interlace 2 and 16 or 18 sector double density format at interlace 3. Don't worry if 18/3 pauses at the end of each track -- this is just the extra revolution forced by having sectors 17 and 0 adjacent on the disk.

When you try this with 80 track drives, don't be surprised if the results are different. The time required for the head to settle into a wide standard track may not be adequate to get it reading properly from the narrow tracks on the quad drive. Such subtleties as erase delays and disk quality are also more critical on the skinny, low power tracks. My Mitsubishi 4853s (96 tpi) will support both 16/3 and 18/3 but are unreliable at 18/4, while my TEAC 55Bs support all three at 48 tpi. Don't take chances with any setup that is marginal. The error rate may be low, but it always seems to happen to a file that isn't backed up.

Hot Rodding

If you want to try for a little more speed, there are two more tricks you can use. The faster WD1772 FDC chip is pin compatible with the standard WD1770 supplied by Myarc. It will try to step the head at 2ms rather than the 6ms setting of the standard chip. (The 80 track EPROM automatically uses the fastest step speed available.) Many of the latest drives can step at 2ms or 3ms even though they are conservatively rated at 4ms or 5ms. The change is noticeable but may not be worth the high price of the WD1772 (it is not a commonly used chip and is rarely discounted). The second fix is cheap and very useful for producing large quantities of copies. The FDC chip's automatic "write verify" function can be defeated by shorting one pin on the controller card to ground. This is best done with a switch so the verify can be enabled for normal operations. The effect of this modification is equivalent of the "turbo" option on the

Corcomp controller and should be used only after testing.

Interlace Patterns

Note: The configurations marked * and ** are the standard interlace patterns for TI and Corcomp formats. The end-of-track intervals are only approximate since the 9 and 16 sector formats include more buffer space than the 18 sector format.

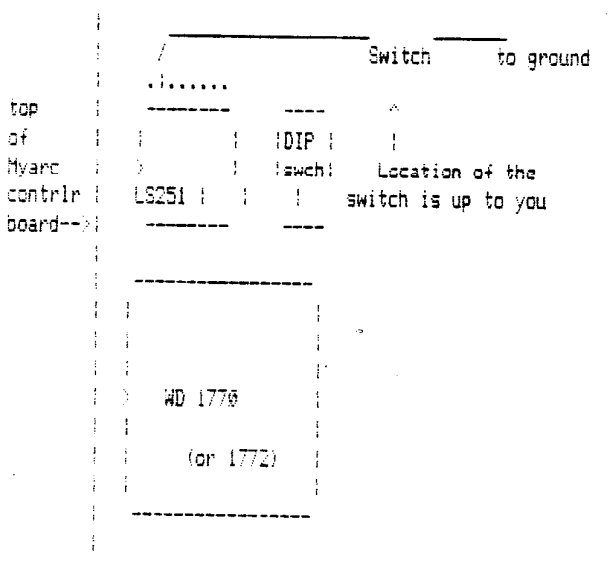
Sect/ trk	Inter lace	Pattern (dashed line is time available for head step)
9	4 *	Ø 7 5 3 1 8 6 4 2
9	2	Ø 5 1 6 2 7 3 8 4
18	5 **	Ø 11 4 8 15 1 12 5 9 16 2 13 6 1Ø 17 3 14 7
18	4	Ø 9 5 14 1 1Ø 6 15 2 11 7 16 3 12 8 17 4 13
18	3	Ø 6 12 1 7 13 2 8 14 3 9 15 4 1Ø 16 5 11 17
		none
16	5	Ø 13 1Ø 7 4 1 14 11 8 5 2 15 12 9 6 5
16	3	Ø 11 6 1 12 7 2 13 8 3 14 9 4 15 1Ø 5

16-sector patterns are not precisely to scale

Disk Controllers -- an Addendum March, 1907

I mentioned a "turbo" modification to lock out the "read after write" (write verify) routine usually performed by the controller. Here are the details:

Find the 74LS251 chip at the top center of the controller board, above the DIP switches and beside the large FDC chip (marked WD1770). Solder a wire from the number 2 pin of the 74LS251 through a switch to ground (e.g. the wide trace of the DIP switches or any trace connected to that wide trace). It looks about like this from the bottom (non- component side) of the board.



As always you proceed at your own risk. (One person has told me this did not work on his 40 track system, but I haven't verified that.) You can tell it is working if your controller writes as fast as it reads (normally the write takes twice as long).

Since I wrote the article on disk controllers, I have discovered some surprising facts about my own system. All of the Myarc timings in the article were done on an 80 track system with the fast WD1772 controller chip (stepping at 2ms). With some help from Paul Charlton and Richard Roseen, I recently customized Paul's Eprom to step at 3ms using the WD1772. (Richard's drives were making errors at the faster speed.) I used a Mechatronics Eprom programmer to download the Eprom code to disk, changed the FDC commands with a sector editor, and wrote the altered code back to a fresh Eprom. The process is simple (and cheap) once you decide what code you need in the Eprom.

The slower step speed made it possible to notice some slight differences in the performance of the WD1772. The first thing I noticed was that interlace 4 on 18 sector tracks was no longer smooth -- it was missing the first sector after a track seek and forcing an extra revolution of the disk. This was the first clear indication of how close this format is to the "ragged edge". The reaction to the small change in step speed implies that this interlace comes within 5% of the minimum time required to step and settle the head. Thus the likelihood of read/write errors is relatively high with this interlace. It will occasionally detect the sector ID and begin to read or write before the head has completely settled. This interlace should definitely be avoided -- 18/3 is both faster and more reliable.

The Eprom modification itself was an interesting experience. I patched the new FDC commands into some unused text bytes and patched addresses into the code to point to the new locations. The Mechatronics Eprom Programmer is an excellent piece of equipment. It will burn (program) a 2764 (8k) in about 90 seconds using the fast algorithm. I have talked to Jim Horn and Jeff Guide about offering an Eprom service to the customers of Disk Only Software. There are many possibilities this technique opens up. There is the 80 track modification for the TI controller worked out by Andy Cooper. And many Myarc owners are still using old Eproms that have never been upgraded (though this situation has improved since Lou Phillips increased his production capacity). The fix we developed for Richard's controller can provide the optimum step speed (2ms or 3ms) for different disk drives using the WD1772 FDC chip. And any enterprising programmer can get his tailor-made code installed in nonvolatile memory.

Jerry Coffey [74716,35251]

HIGHLIGHTING

By Earl Raguse

Last month at the BUG meeting, during a demo of EZ KEYS, at least that's what I think it was, a feature was demonstrated that allowed one to switch the forground colors of certain characters to make them stand out from others like O vs 0, and 1 vs l, or for trouble shooting of bad typing, something I do real well. Aha, you say, I can do that with CALL COLOR. True, but its not permanent. I don't like having to embed trouble shooting routines in my programs if there is an easier way.

The following program called HIGHLIGHT makes permanent foreground/background color changes and can be controlled ON and OFF at will. Once executed, the program can deleted with NEW before you start entering a new program. I some times put this in my LOAD program, its easy to turn off if you don't want it.

I found the basic program idea in the Tacoma 99ers Newsletter of December 1987, the article was by Joe Nolan, who credits Harry Wilhelm of the Twin TIers UG with the original idea. I don't have any idea how much evolution has gone on, but I added my two cents also.

Lines 130 and 140 do all the work, and if you wish to transfer this effect to one of your own programs, that's all you need. The following tells you how you can change these lines to suit your needs. If you study it a bit, you can see the potential for other purposes.

In line 130,

(1) Change the eighth number, from the address, 17, to the number of the first character set you want to change PLUS 15, The current program is 15+2=17 for character set 2.

(2) Change the eighth number after that, 3, to the number of character sets to change. The current program is 3 for character sets 2, 3 and 4.

In line 140,

(1) Load a number, (in this case 244) for each character set to be changed. That number is computed as (16*(FG-1)) + (BG-1) where FG and BG are the Foreground and Background color numbers as defined in the XBASIC manual. Each character set could have a different combinaton of colors. The program as written is for all characters white on blue, ie (16*(16-1)) + (5-1) = 244.

(2) The effect is turned ON by CALL LOAD(-31804,63) and OFF by CALL LOAD (-31804,0). This can be done either in a program or from the keyboard. I added the lines 150 and 160 for easy control of the effect on or off. These can be deleted if not wanted.

```
100 ! SAVE DSK1.HIGHLIGHT
110 !By Joe Nolan, Tacoma 99
ers UG Newsletter Dec 87,
Original idea by Harry
Wilhelm of Twin TIersUG
120 !Modified by E Raguse
UGOC 1/87
130 CALL INIT :: CALL LOAD
(16128,2,224,38,0,2,0,8,17,2
,1 ,63,36,2,2,0,3,4,32,32,
36,2,224,131,192,3,128)
140 CALL LOAD(16164,244,244,
244 ::CALL LOAD(-31804,63)
150 PRINT "TURN IT OFF? PRES
S SPACE,   ELSE ANY"
160 CALL KEY(0,K,S):: IF S=0
 THEN 160 ELSE IF K<>32 THEN
 END ELSE CALL LOAD(-318
04,0)
```

APRIL DEMO

By Curt Borders

LOAD INTERRUPT, HOLD AND RESET SWITCHES FOR THE TI-99/4A COMPUTER

LOAD INTERUPT:

THE LOAD INTERUPT, WHEN ACTIVATED WILL CAUSE THE COMPUTER TO SUSPEND IT'S CURRENT OPERATIONS. THEN IT WILL LOOK AT A SPECIFIC MEMORY LOCATION THAT WILL TELL THE COMPUTER WHERE TO GO FOR THE NEXT SET OF DIRECTIONS.

HOLD:

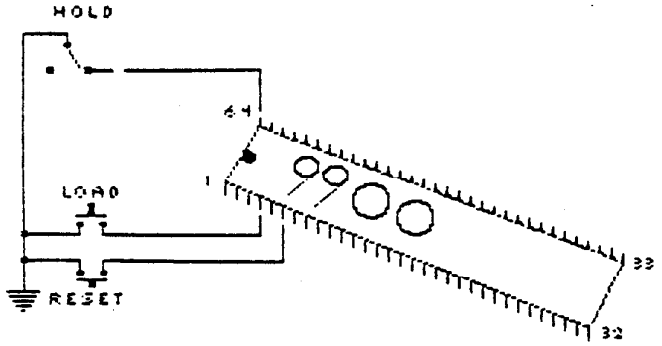
THE HOLD DOES WHAT IT IMPLIES. IT PUTS THE MICROPROCESSOR ON HOLD. IT'S GOOD FOR STOPING THE COMPUTER DEAD IN ITS TRACKS WORKS GREAT FOR GAMES THAT DO NOT HAVE A PAUSE FUNCTION. THERE ARE TIMES WHEN YOU DO NOT WANT TO USE IT, LIKE DURING A DISK READ OR WRITE OR INITILIZATION ROUTINE.

RESET:

IT CAUSES THE COMPUTER TO DO THE POWER UP ROUTINE. THIS IS GREAT WHEN THE COMPUTER LOCKS UP. YOU HIT THE RESET SWITCH AND YOU ARE BACK TO THE TITLE SCREEN. THIS SAVES WEAR AND TEAR ON YOUR POWER SWITCH AND EXTENDS THE LIFE OF THE COMPUTERS POWER SUPPLY.

PARTS

1-SP/ST MINI SWITCH	RS# 275-620 OR 275-1545
2-MOMENTARY SWITCH	RS# 275-1547



Did you know that...?

by Chick De Marti

May 1988

SPICE UP YOUR CALL KEY ROUTINES
* Ollie Hebert *

How about a bouncing question mark for a prompt? As listed below, this mini-program will demonstrate the idea with only a little typing on you part. If you like it, incorporate it into your own program.

```
150 DISPLAY ERASE ALL :: CAL
L CHARPAT(63,C$):: CALL CHAR
(64,SEG$(C$,3,14))

300 DISPLAY AT(4,3):"SELECT
FROM THESE ITEMS:" :: FOR I=
1 TO 4 :: DISPLAY AT(I*2+7,8
):CHR$(I+64);":  PROCESS #";
CHR$(I+48):: NEXT I
310 DISPLAY AT(17,8)BEEP:"E:
  QUIT": : : :"    YOUR CHOI
CE (A-E)  ?" :: GOSUB 10000
:: ON X-64 GOTO 1000,2000,30
00,4000,5000

1000 PRINT X-64 :: GOTO 5000
2000 PRINT X-64 :: GOTO 5000
3000 PRINT X-64 :: GOTO 5000
4000 PRINT X-64 :: GOTO 5000
5000 END

10000 Z=64+(Z=64):: DISPLAY
AT(21,24):CHR$(Z):: CALL KEY
(3,X,Y):: IF Y=0 THEN 10000

10010 IF X<65 OR X>69 THEN C
ALL SOUND(200,220,0):: GOTO
10000 ELSE CALL SOUND(200,14
00,0):: DISPLAY AT(21,24):CH
R$(X):: RETURN
```

How about altering the color of the question mark? Only two of the above statements need to be changed. (lines 150 and 10000)

```
150 change CALL CHAR to CALL COLOR(14,12,1)
                     :: CALL CHAR(143,C$)
  1000 change Z= TO Z=143+80*(Z=143)
```

This little gem found in SMAUG/99 newsletter

~~~~~~~~~~~~~~~~~

WHAT DO YOU MISS?

By not showing up at a meeting, besides not seeing Steve Mier's demo of MINDEX (Database) and Ray Kazmer's new load program that will, "set off autoalarms and open garage doors for a 5 block area"...you also miss some good ... fellowship (or is it personship?). Ray, Steve and I got to talking, and I learned, "You can use a semicolon to replace a SIZE command as per this demo" :

```
100 CALL CLEAR :: CALL COLOR
(9,5,5):: FOR C=1 TO 32 STEP
 2 :: CALL VCHAR(1,C,96,24):
: NEXT C
110 DISPLAY AT(3,1):"THIS IS
 TEST A" :: DISPLAY AT(6,1)S
IZE(14):"THIS IS TEST B"
120 DISPLAY AT(12,1):"THIS I
S TEST C"; :: DISPLAY AT(15,
1):"BY RAY KAZMER"
130 GOTO 130
```

I mentioned (as we had our coffee and doenuts that some times it would be nice to have 2 or 3 BEEPS as a warning, but as you know. You can't enter: DISPLAY AT(1,1)BEEP BEEP BEEP. So Ray sez, "Oh yean? Write it this way:"

```
125 FOR A=1 TO 3 :: DISPLAY
AT(1,1)BEEP:" " :: CALL SOUN
D(300,4E4,30):: NEXT A
```

(These last two items were written on a paper napkin...thanx again Ray)

Not to be out-done, line 125 can also be:

```
125 FOR A=1 TO 3 :: CALL SOU
ND(200,1400,3):: NEXT A
```

And here's a little surprise. Enter this exactly as I have it typed:

```
NEW
10 CALL CLEAR
20 CALL COLOR(9,5,5)
30 CALL HCHAR(5,5,96,5):: HC
HAR(6,5,96,5)
40 GOTO 40
```

RUN this...then figure why it do what it do?

~~~~~~~~~~~~~~~~

Another good reason for getting involved in your club meetings, SIGs, commitees, etc.

```
++++++++++++++++++++++++
|  I HEAR, I FORGET    |
|  I SEE, I REMEMBER   |
|  I DO, I UNDERSTAND  |
++++++++++++++++++++++++
```

See you at the next meeting? I'll be there!

~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~

(Did You Know ... cont.)

3.5" Toshiba Floppy Drives

In March I have a short article about the availability of this product. I received this review from member Ralph Jones of Morrow Bay.

3.5" Toshiba floppy drive conversion for the TI/994A available Alpha Scientific described by Mr. Leo D Ross - 7710 40th Ave E, Tacoma Wa 98443.

The drive is 1/2 ht., it fits right into the "P"box, comes in two packages, the drive, and the mounting hardware. All that is needed is a screw driver. It is double sided and can be formatted by the TI disk controller as a single sided or double sided (90k or 180k). Corcomp or Myarc controllers will format to ss 360k or ds 720k. I had two of the 3.5 drives installed in the P box, but found it awkward it use them in the vertical position, so I moved them to an external box. I now have a 5.25 drive and two RAM disks in the P box, and the 3.5 are in the external drive box. I number them #1 3.5, #2 5.25, #3 3.5, #4 Ram disk#2. Ram disk#1 can be assigned any drive number between 1 and 6. When assigned a number where there is a drive, such as #1, the 3.5 drive becomes transparent (all commands to drive #1 are answered by the Ram drive#!). I am very satisfied with their use. Two main advantages are mechanically superior disk, and much less storage space.

Thanx Ralph Jones for sharing this with us.

~~~~~~~~~~~~~~~~

More on RND numbers

To continue our discussion about TRUE random numbers. Try this short routine:

```
10 FOR I=1 TO 10
20 R=INT(RND)+1
30 PRINT R;
40 NEXT I
```

RUN this routine 4 or 5 times and you will get the same sequence of "random" numbers. For a true RND number, put the request within a CALL KEY statement. Because not all people will press a key at exactly the same time the computer will read down it's list of pseudo-random numbers an unpredictable number of places. Now try it this way:

```
10 CALL KEY(0,K,S)
20 R=INT(RND)+1
30 IF S=0 THEN 10
40 PRINT R;
50 GOTO 10
```

(By the way, Tom Freeman uses this type of a random routine for his Raffle drawing).

~~~~~~~~~~~~~~~~

COLISTER A TINYGRAM by Ed Machonis

Not just another 28 column lister! "COLISTER has a couple features not available in other programs. First, it prints a blank line between program lines...makes it easier to read. Second, it TABs the output 6 spaces (centering the program in an otherwise 40 Col listing). COLISTER does not require a program be resequenced! And COLISTER will print to either disk or printer. Don't print the listing through the formatter unless you have modified your Formatter to ignore the special format command characters that are also found in programs. To use COLISTER:

```
1 !  *** COLISTER ***
 A Tinygram by Ed Machonis
   QB-99ers, Bayside, NY

2 PRINT :"1st LIST your prog
rm to diskThen RUN COLISTER"

3 PRINT :::"INPUT FILENAME?
ex:DSKnLIST" :: "INPUT F$ ::
 INPUT "OUTPUT FILENMAE? ex:
 PIO OR DSKn.LIST28 ":P$

4 OPEN #1:F$,INPUT :: OPEN #
3:P$,OUTPUT :: ON ERROR 10

5 C=28 :: LINPUT #1:A$ :: IF
 LEN(A$)<80 THEN 8

6 LINPUT #1:B$ :: IF VAL(SEG
$(A$,1,POS(A$," ",2)))<VAL(S
EG$(B$,1,POS(B$," ",2)))THEN
 F=1 :: GOTO 8

7 A$=A$&B$ :: IN LEN(B$)>=80
 THEN 6

8 A=LEN(A$):: L=A/C+.99 :: F
OR I=0 TO L :: PRINT #3:TAB(
6);SEG$(A$,1+I*C,C):: NEXT I
 :: IF EOF(1)AND F=0 THEN CL
OSE #1 :: CLOSE #3 :: END

9 IF F=1 THEN F=0 :: A$="" :
: GOTO 7 ELSE 5

10 ON ERROR 10 :: RETURN 7
```

(Thanx Ed Machonis and the QB MONITOR).

~~~~~~~~~~~~~~~~

Well, I'm out of coffee. See you next month

Chick
~~~~~~~~~~~~~~~~

E Z * BASICS

by Chick De Marti

EZ-BASICS (3)

Welcome back, gang. The last two months have been prepatory lessons. KInd of like learning the keyboard before you sit down to play your first tune. While I do want this tutorial understood by every (the swift and not so swift), I don't want to give my students reason to yawn. With that, let's get to it.

Last month, I promiced we would learn about variables, INPUTing, and LOOPS. This is the logical sequence when trying to understand loops.

* VARIABLES *

There are two types of variables, numeric and string variables. Examaples might be:

```
N    ( a numeric variable )
N$   ( a string variable  )
```

As you can see, you can tell the difference between the numeric and the string variable by the ($) dollar sign (kind of looks like an "S" for String). Let's look at some exams.

```
N=4          N$="FOUR"
NUMBER=10    NBR$="NUMBER"
FLAG=0       MSG$="Press any key"
```

Three important rules:

1. Use meaningful words for variables.
2. Don't use words that are researved by TI suc as LIST or NUM.
3. The values of a string variable will always be within quotation marks!

NOTE: Because of rule #3, a string variable CAN NOT be mathamatically acted upon.

```
Exam. 100 A$="4"
      110 A$="4"+3     ( A no no !)
```

NOTE: A little publicized fact..."A string-variable INPUT will accept either a number or a letter as a responce."

```
Try   100 INPUT "PRESS ANY KEY ":TEST$
      110 PRINT TEST$
      120 GOTO 100   (use FCTN 4 to quit)
```

Press either a number OR a letter, or even just press <ENTER> or the return key. Works O.K., huh? Now take the ($) dollar sign away from lines 100 and 110 (making TEST a numeric variable) and RUN the program again.

Gee, if a string variable will accept either type of INPUT, why don't we use the string variable at all times? (If you don't know, study the example above (A$="4")).

Befor we go on, let's try a true/false test on the eligibility of various variable names.

```
1. SECOND=35          5. RND=50
2. RES=15             6. RANDOM$="50"
3. HOUR="10:15"       7. MSG$=4
4. BAL="TEN"          8. DIM=20
```

(Answers on next page)

Now that we have a pretty good idea of what a variable expects, let's put them to use.

* INPUT *

Format is INPUT "optional msg.":n (n$)

Let's try out our new found knowledge...

enter NEW (to assure there is nothing in memory!)

```
10 CALL CLEAR
20 INPUT "Enter a name":NAME$
30 INPUT "Enter an age":AGE
40 PRINT : :NAME$;AGE
50 INPUT E$
60 GOTO 10
```

Line 10 clears the screen (the "NEW" command I won't bother to explain).
Line 20 expects a "STRING" variable.
Line 30 will ONLY accept a numeric value.
Line 40 uses 2 colons BEFORE the string variable to give us 2 blank lines (Remember EZ-BASIC(1)?) between the INPUT statement and the PRINT command. The semicolon in this line places the AGE next to the NAME.
NOTE: TI automatically places a space before (and after) a printed number.
Line 50 Here we used an INPUT statement, without a message, as a PAUSE routine... You must press enter to continue... (notice NO COLON required if there is no message!)
Line 60 Let's do it again.

O.K. - I guess we are ready for...

* FOR...NEXT *

Format FOR character=n1 TO n2 (where n1 is the starting variable, and n2 is the ending variablle, and character is the variable you want the LOOP assigned to.)

The FOR...NEXT statement is one form of creating a LOOP. A LOOP is an action you that you want performed a certain number of times. For instance:

```
        10 FOR X=1 TO 10
        20 REM ...an action...
        30 NEXT X (this variable must match
                   the one in line 10)
```

```
Exam.   10 FOR COUNT= 1 TO 10
        20 PRINT ,COUNT
        30 NEXT COUNT
        40 END
```

In this Exam. I used COUNT as my variable, or loop name. It's always a good idea to use a meaningful name for your variable. Such as LINE,SPACE,NUMBER (again... DON'T use T.I.s reserved words). This is easy loop to figure out. 10 sets up a COUNT from 1 to 10.

20 PRINTs the current "COUNT" (did you notice the (,) comma? Last month we learned that the comma TABs the output to the center of the screen (or T.I.s 2nd column).)

30 is the required NEXT, the 2nd part of the FOR...NEXT command. And finally

40 END (not required but a good idea).

Want to see something interesting? Add another comma to the end of line 20, ie:

```
20 PRINT ,COUNT, (Before you run the
                  prog. WHAT WILL IT DO?
```

Here are a few useful FOR...NEXT loops.

```
++++++++++++++++++++++++++++++++++++++
|                                    |
| Yes Nancy! There are other, more ef- |
| fecient methods to do these things - |
| but these will come in their logical |
| order.  Hang in there, gang!  Chick  |
|                                    |
++++++++++++++++++++++++++++++++++++++
```

#1. A delay loop...used:

```
10 DX=600   (length of delay desired)
20 GOSUB 1000
1000 FOR DELAY=1 TO DX
1010 NEXT DELAY
1020 RETURN
```

NOTE: Obviously, XBasic can put the routine on one line...ie:

```
1000 FOR DELAY=1 TO DX :: NE
XT DELAY :: RETURN
```

#2. Center a title...used:

```
30 PRINT TAB(8);"MY PROGRAM":
 : :TAB(6);"by Chick DeMarti"
40 RX=14    (The ROW you want the
             BOTTOM line to be on.
50 GOSUB 2000
60 GOSUB 1000   (use #1. DELAY loop).
2000 FOR ROW=24 TO DX STEP-1
2010 PRINT
2020 NEXT ROW
2030 RETURN
```

NOTE: XBasic can create the same effect with out the subroutine. Use:

```
30 DISPLAY AT(11,8):"MY PROG
RAM": : :TAB(6);"by Chick De
Marti"
```

#3. A Vertical WIPE...used:

```
70 TOP=11
80 BOT=14
90 GOSUB 3000
3000 FOR COL=1 TO 32
3010 CALL VCHAR(TOP,COL,32,2
4-BOT)
3020 NEXT COL
3030 RETURN
100 GOSUB 1000 (another delay)
999 END   (Try running Prog. without it)
```

++++++++++++++ Answers to test ++++++++++++

1. true
2. Illegal Comm. (RES is reserved)
3. string-number mismatch
4. string-number mismatch
5. syntax error (RND is reserved)
6. true
7. string-number mismatch
8. syntax error (DIM is reserved)

Well there you have it for this month. As I promised, you were introduced to VARIABLES... INPUT,GOSUB and FOR...NEXT loops. You're becoming more comfortable with TABs,commas, and colons. But we've only scratched the surface of the powerful FOR...NEXT command. Experiment with some of the things we have done... (Try making the WIPE go from right to left!)

Next month we'll continue with our FOR-NEXT loops and start on a program (no not a game!) (unless I find one with neat routines). Till next month, remember, "YOUR COMPUTER LOVES YOU" ... spend some time with it!

ARTIST FUNCTIONS

SYMBOL	FUNCTION	TYPE	USE OF ENTER/FIRE	COMMENTS
D	Draw	cmd	start/stop	hold fire down
P	Point	cmd	place	
L	Line	cmd	begin/end	
K	K-line	cmd	begin/intermediate	D to exit
R	Rays	cmd	center/start/stop	D to exit
F	Fill	cmd	do	space to abort
V	Frame	cmd	1st. corner, opp. corner	
X	Box	cmd	1st corner, opp. corner	fills w/ pattern
O	Circle	cmd	center/edge	
G	Disk	cmd	center/edge	fills w/pattern
CTRL-A	Clear Image	cmd		leaves color, pattern
H	Hor/Vert	cmd	begin/end	
W	Swap	cmd	new color/old color	
I	Invert	mode		negative image
E	Alpha/Numeric	cmd	begin loer left/stop	does not use fonts use CTRL x for width FCTN x for height
CTRL-B	Clear Color	cmd		leaves pattern
S	Store	menu		load/save/index picture
Z	Zoom	mode	select window	move with cursor cmds
M	Mirror	mode		4 reflections
C	Hard Copy	menu		
FCTN-.	Plot/Erase	mode		
FCTN-,	Foregnd Color	cmd	change to next color	
FCTN-;	Cursor Speed	mode		fast/slow toggle switch
F icon	Foreground Background			color chosen will be foreground or backgrnd
P icon	Pattern in use	mode	display next pattern	only P is solid
- icon	Color Cursor	mode		use P to clean up color borders

ENHANCEMENT FUNCTIONS

M	Move w/o color	top left/bottom right/place		T to check
N	Move w/color	top left/bottom right/place		T to check
C	Copy w/o color	top left/bottom right/place		T to check
D	Copy w/color	top left/bottom right/place		T to check
A	Alpha Numeric	menu	enter/place text	SPACE to exit T to check
	Use a Slide		select place	SPACE to exit
S	Slides	menu		
	Define		pick box/define slide	SPACE to exit
	Erase		pick slide	SPACE to exit
	Rotate		pick slide	SPACE to exit
	Load Slide file			_S added to filename
	Save Slide file			_S added to filename
	Load Instance		enter name/place instance	T to check _I added to filename
	Save Instance		enter name/select top lft bottom right	_I added to fulename

keyboard cursor movement:
Horizontal/vertical--S/E/D/X
Diagonal--W/R/C/Z

Reprinted from the Dallas User Group!

How to OPERATE ... or doctor things up

by D. R. Fudge

Two months in a row? Unheard of! Even the author is shocked at this turn of events. In spite of the fact I own and use John Johnson's very useful "Remind Me" program I never log the deadline for the next newsletter article and successfully surprise myself each month that once again it's time for my next assault on everyone's intelligence. This month, I so surprised myself that I am going to do a multiple review and rave on about several programs that I find myself using over and over. You almost certainly have seen excellent reviews of most or all these programs, but probably have never seen them abused in my particular style. I have meticulously selected these programs by looking on the top of my PEB and disk drive #2 container (proudly home built and ugly!). If I use a program often, it usually gets left in one of these two places, which brings me to a new caution that should be included on the sleeve of every floppy disk in the universe! Do not, repeat, DO NOT! allow your cats to sleep on your PEB if you store your floppies flat on top of it!!! I am now the proud owner of some disks with errors and dead files because my cat found it necessary to soften its sleeping place with it's claws, and punched minute holes in the magnetic media, and in at least one instance, completely through the disk.

Now, on to the mini-reviews. The first, and obvious, and reviewed to death choice is Funnelweb. The very existence of this article depends heavily on my use of the TI-WRITER portion of that disk. I use that particular word processor because I have the module, manual, Funnelweb based version on disk, and am (finally) completely at home with its format. By using the commands in the manual, various transliterations, the control U codes (lists of the last two have been published by myself as well as many others) I can use any printer command at my disposal on my TI impact printer. Also, I can type on screen in 33 columns and listen to Mike Dodd gripe because it looks funky on his 80 column display via his 9640 when I upload it to him for sending to Terrie and/or Tom. Funnelweb, as you know doesn't stop at word processing (in fact, that is a mere beginning). It also allows me to load in DM1000 for very extensive file handling, and many other features best handled in a full blown review. One more I will mention is the loader menu which allows me to fire up in Extended Basic and autoload Funnelweb, then when I need to go to something else which happens to be in Assembly language, I don't have to kill everything to load it. With the rapid spread of Extended Basic loaders, this feature is getting used less often, but it is reassuring to know it's there when I need it!

The next obvious choice of programs is the terminal emulator. As I said, I usually upload this column to Mike Dodd, and so that is one of the reasons for using a terminal emulator. The only problem here is which one to use. I think there are several around which have their own unique features to make them valuable in their own right. Three of these are: Mass Transfer, Fast Term, and Telco. I honestly can't say that any of these are better (totally) than the others, but each is worthwhile for it's own reason. Right now I usually reach for Telco because it's handy (on my PEB) but that is subject to change should I ever become organized enough to actually have a real place to put things.

As I have already mentioned, one of my most used programs is "Remind Me!". Mostly, I used it to keep up with what shift I was to be working at any given time (subject to change in a heartbeat and at the whim of my supervision), but now that I'm on straight day shift, I use it for three main purposes. (1)To keep track of my scheduled vacation, and when I used it instead, (2) to keep track of whose birthday I forgot last month and what holidays are coming up, and (3) When the next Street Rod event is coming to pass, and where it's being held. About the only added useful feature I could suggest for my needs would be an alarm feature to remind me that it is time to shut down the computer and go to my scheduled whatever, although that would be reasonably difficult to use on a 99/4A without a real time clock available. Also, it would be nice if I could remember to mark the deadlines for these newsletters some way so I wouldn't have to be interrupted on page 35 of a script for **STAR TREK: THE NEXT GENERATION.** No, I don't live next door to Gene Roddenberry, nor do I work for a television producer. If I may digress for a moment, (or if not) the current strike by the Writer's Guild has had a unique and valued fallout for Trekkies such as myself and Mr. Dodd. One of the script authors had a (to me) wonderful idea for raising funds for the strike. He offers a script, written by himself, unfilmed for television of an episode of the aforementioned program, for a donation. How did I find out about this? Thanks to Mike Dodd, terminal emulators in general, and Compuserve's Science Fiction Forum, upon which the author placed the offer. Also, it allows me to watch the series, via my satellite dish in crystal sharpness, and then rewatch each episode via network TV. Aren't TE's wonderful for finding such valuable information?

Next on my list of often used programs are two programs which have their own special features for particular needs. These are ADVANCED DIAGNOSTICS and OSKU. These are especially valuable to me with the free

roamings of my cats, my young'uns, and myself. It's almost a certainty that one of the three categories of us will cause some mishap to a disk sooner or later. Thanks to these two programs, I am often able to recover, and/or repair, the file/s. Also, I can learn more about the disk itself, and even make changes to assembly language programs to suit my particular situation. A good example of which is the @ and % in the TI-WRITER portion of Funnelweb. I use either program to change these symbols to utilitarian symbols rather than formatting codes thanks to the Funnelweb authors and many others.

Lastly, (because of time and space) there is DM99. My, isn't it handy to have a resident disk manager there for the asking? That Dodd character occasionally comes up with something worthwhile in spite of my interference. There are many other disks I use -e.g. Mcopy, TI-Artist, various games, etc.- but I think I have briefly covered the ones I use most often. Actually, the ones I use most often are uncovered. Get out of here, cat! I'll feed you to the disk drive if you mess with me!

BEGINNING FORTH #1

by Earl Raguse, LA 99ers

This is a trial balloon for a series of future articles on the wonderful language of Forth. After talking with Terrie Masters and Chick De Marti, I decided that there are TI99/4A users who are still interested in learning Forth, but just haven't started for one reason or another. One reason seems to be that some good writers have started out with a simple tutorial format, then got bored or something and quit before any of the readers really got a handle on Forth. The goal of this series is to change that.

Some of you may be aware that I have been writing Forth articles in the Orange County UG ROM for the past two years. That series was a chronology of my learning process, and I took many wrong turns, and am still in the learning process. I have learned enough however that I hope not to lead you down any blind alleys, at least not until we get to more advanced stuff. Then you will have to take your chances with the rest of us.

There aren't many sources of teaching information on Forth, in particular for the FIG 79 version of Forth, which is what TI Forth is. FIG stands for Forth Interest Group, and Forth 83 is the current version.

The one outstanding book on the subject was STARTING FORTH, by Leo Brodie, which was excellently written and had humorous cartoons to explain the working of Forth words. This book was really intended to teach Polyforth, the Forth language implemented by Brodie's company, and not all of it was compatible with TI Forth. The TI Forth Manual, however, has an Appendix which points out the differences in the two Forth versions. Unfortunately for TI Forth users, Brodie has rewritten his book to be compatible with Forth 83 and thus doesn't match TI Forth anymore. The first edition is no longer available in bookstores, but can be found used at reasonable prices.

Forth is not a dying language as I have heard in certain rumor mills. Forth is used in the development stages of almost all new computers, it is usually the first language implemented. This is because Forth is can be written more quickly and is easier to modify, as compared to competing languages such as Assembly Language (A/L), Pascal and "C", very important during development, but C is usually used to write the mature operating system because of its speed. Forth is used to write Word Processor programs, Artificial Intelligence programs (AI) and operating systems. Forth is one of the top languages used for Real Time Control. Forth is a compiled language, hence it is fast and is transparent to the user.

Major strengths of Forth are:

(1) Its extendable compiler, more on that later.

(2) Ease of testing pieces of code as it is being written. Each word is executable by itself, no driver program is required. Only LOGO compares in this respect.

(3) Ease of making revisions. Changes in one word do not relect in others. Basic, for instance, is very difficult in this respect, as are most of the high level languages.

(4) Very powerful, in that Forth has total access to the machine's capability. If it can be done in A/L, in can be done in Forth.

(5) Forth is fast, only machine code is faster. If Forth doesn't do a particular operation fast enough, that operation can be written in assembly language and compiled by Forth's built in Assembler. Linking is automatic and transparent to the user.

(6) Well written Forth is very readable. Of course there are programmers who seem to delight in producing "Write Only" Forth, unreadable by even themselves after a short period away from it. The trend is away from this however.

There are other reasons for learning Forth, but you get

the idea. There are some initial disadvantages, of course. Aren't there always?

(1) One must learn Reverse Polish Notation, and LIFO stack operation. The average TIer masters this in 93 minutes.

(2) One must learn a totally different approach to programming. This is usually called "Top down design and bottom up coding". This is not really a difficult obstacle, but it is easier for those who never learned BASIC or FORTRAN.

(3) Forth does have to reside in memory (its not a TI module) and hence restricts program size. Forth is very compact however, and thus this usually is not a limitation even though the 99/4A is memory starved. For those of you who go on to be experts and write huge Forth applications, there are Super Carts to hold Forth and thus leave TI memory for programs and data. RAM disks can be used to advantage also if you partion your code into related packages.

SO WHAT ARE WE GOING TO LEARN HERE?

My plan is to start with the very basics, then continue to the limit of my Forth knowledge, see my learning plan outline below. I will assume only that you know where the power switch is, that you know how to operate your computer and that you have all necessary equipment connected and turned on.

WHAT IS THE NECESSARY EQUIPMENT?

Well, as a minimum you must have a console with monitor, 32k memory expansion, disk controller and at least one disk drive (I will only briefly touch on single drive operation, its a drag). You also must have a way of loading Forth. That can be done using Editor/Assembler or FunlWeb(Writer). There is a Forth disk available with a machine language loader included on the disk for access by XBASIC; if its not now in the library, I will put it there. You can learn Forth without having an RS232 interface and a printer, but it sure isn't much fun.

You must also have a TI Forth Manual (TIFM). I don't intend to parrot that back to you in its entirety. There is much it doesn't say, but what it does say is reasonably clear. Where I think it warrants, I will explain it as I see it. If you don't have one check our library, or talk to Chick De Marti.

LEARNING PLAN

1. Getting started, some terminology, the screen concept. fixing TI errors, making BSAVED Working Disks and Data Disks. Elementary words, printing text to the screen and printer.

2. Fixing the Editor to have an auto repeat cursor. You can't appreciate this unless you have done without it.

3. Stack manipulations and doing integer arithmetic and logical operatons.

4. Learn how to define new words and use loops, both definite and indefinite.

5. Rewriting some Forth words to suit yourself and making some new ones. Better known as making life easy for yourself. Know where not to change Forth.

6. Floating Point Arithmetic (FPA) and how to live without it, unless you are desparate and not in a hurry.

7. Trignometric functions etc. without FPA.

8. Graphics capability of Forth and some simple and not so simple examples.

9. Generating sound in Forth. There are no resident sound control words, but no matter, we will make our own, and some music too.

10. String handling words that we will define. Forth has very few resident ones, but we fix that.

11. Files and disk access, writing screens under program control. Elementary sorting with an animated demo.

12. Subscripted variables, multi- dimensional arrays and matrices. How to use compiler extension ability.

13. Making Forth talk, anything the 99/4A can do Forth can do.

14. Other stuff like Reverse Video highlighting, special requests etc.

WARNING-FEEDBACK REQUIRED

The above plan can and will be changed on request, if its reasonable and I know how. I will also probably change it to account for things I have forgotten, or new things I learn. I'm always doing that.

Because of the time lag between writing of these articles and printing The TOPICS, three months of these articles will be prepared, if the author or the TOPICS editor does not receive an indication of interest in Forth, the author will know that he is beating a dead horse. If there is in fact interest, the articles will continue as long as you want them. I have been writing about Fourth for two years now and don't intend quit on you.

You may call me, Earl Raguse, almost any day or evening at 714/847-5875, or talk to me at meetings. If I can't help you directly, I may know someone who can.

FROM THE DISK OF ....

Mike Dodd

I'm baaaaccccckkkk.... the hacker who refused to die, despite the most fervent wishes of many people all over the world! But I'm still here. HA!

I really should be organizing my room, but I <u>really</u> don't want to, so I'll do this instead. This should at least make Terrie happy, although it may not do much to improve relationships with my parents. You can't win 'em all, I guess (one could argue that there are days that it appears impossible to win ANY... but I won't get into that right now).

I just got back from Wisconsin and Chicago! Far out. In WI, I went to a Wisconsin Parents Association conference on homeschooling. I, along with the rest of my siblings (there are three others) are all homeschooled. Another famous (in the TI community) homeschooler is John Calvin Traver, son of Barry and Sharon. So I'm not the only weirdo around here! The conference was great, totally unlike any TI faire I've ever seen (translation: the WPA conference was organized. A terrifying trend, if you ask me.) My entire family went, minus Dad, who was forced to stay home and do the work which Boeing so desperately desires of him <sigh>. In Chicago, I went to the Shedd Aquarium, the Addler Planetarium, and the Cubs vs. the hated San Francisco Giants (Cubs won). I had a lot of fun, and I even have something to constantly remind me of my day at Wrigley Field - a scorching sunburn <sigh>. But I had fun. What does all this have to do with computers? NOT ONE DARN THING. But I need something to write.

Actually, I did get to spend a couple of days messing around with <shudder> an IBM AT. It even gave me a few vague ideas of possible program ideas for the TI. So maybe the trip wasn't a total loss to the TI community, after all. Then again, knowing my programming skills, maybe it was. It all depends on your point of view, and personal bias.

When I returned home, I faced one of the most horrible nightmares known to computer owners: <start humming tune to The Twilight Zone> ... putting the computer back together! While I was gone, Dad painted and floored my room. This meant that the computer, not to mention all the bookshelves and assorted materials (read: more printouts, books, and other misc. garbage than I thought would fit in my room) had to be packed into boxes. It was a rush job, so when I got back, the computer was scattered amongst my closet, the living room floor, and about a dozen boxes. Y'know, this computer has more %*$!^ cords and cables than I have ever seen in one place before in my life! So it took two days, but it finally came back together.

I suppose I should write <u>something</u> relative to the TI... what I've written so far bears a striking resemblance to what Mike Wright was talking about when he mentioned all the irrelavent, meaningless, useless exchanges going on many of the BBSes. Sorry, Mike, and all others. Also, if you've stuck with me this far, you really do deserve something. Those who have gotten (justifiably) fed up with me and have stopped reading by now can miss whatever pearls of nonsense I happen to drop in this ravings.

Thanks, Tom, for writing the mini-mod to Super XB. What Tom failed to mention is that he had been asking a certain party, who, by the way, wrote part of the SXB code, to do these modifications for quite some time. This person could, I'm sure, have done it in a matter of hours, seeing as how this person had all the source code to SXB, something that Tom did not. Whoever that person is should publicly apologize to Tom, I think.

All right, then: <u>Sorry</u>, <u>Tom</u> <yap, Tom asked me to do it, and I could have, but somehow never got around to it... sigh>

One minor detail I noticed about Tom's article (SEE, I told you I'd get to something relavent eventually). He explains perfectly how to make the modifications by using the Gram Kracker built-in memory editor. Unfortunatly, Geneve owners can't use the GK editor. It doesn't work. So what you need to do is modify the code with some other memory editor, such as Edgar Dohman's (sp??) SBUG II package - use it to modify the GROM addresses, then use J. Peter Hoddie's CSAVE program (which is supplied on one of the MYARC disks) to re-save the cartridge back to disk. If you don't have SBUG, or some other type of editor, you can use a sector editor. The changes will all be in the last saved XB file. If you saved your cartridge under the name of SXB, the changes will be in the file SXB5. There are four addresses where Tom has changes - these translate into:

>6B2A starts at sector 11, byte 48 (hex >30)
>6BB2 starts at sector 11, byte 185 (hex >B9)
>6AD7 starts at sector 10, byte 221 (hex >DD)
>7796 starts at sector 23, byte 156 (hex >9C)

The sector number is the relative offset into the file - e.g. sector 11 is actually the 12th sector of that file.

OK, there's relative piece of info #1 out of the way. What can we write about next .... hmmm.... I seem to be suffering from SEVERE writer's block. It took me forever to come up with something to write about for MICROpendium yesterday, and I'm not doing a lot better for LA today.

Here's an idea: Terrie started the tradition of giving out white, black, and I see recently, grey hats to

various individuals and organizations. In that vein, I would like to give a purple and orange striped hat to D.R. Fudge, who writes some of the most consistently good, funny, articles I have ever seen (he even manages to talk about things relevent to the 99/4A - something I seem to have great difficulty doing), is one of the greatest friends a person can have, and has displayed more patience with my inane (insane?) ramblings and beta-test programs and program ideas than anyone else I know. D.R. is one of a kind. I always meant to thank him like this in K-Town (the Knoxville Users' Group), but I left rather hastily and did not get a chance to. So now, D.R. - thanks!

Well, I'm going to stop. I figure one or two more stupid, boring, insipid articles like this, and Terrie & Tom will stop asking me to write articles for TopIcs. In fact, maybe they'll start asking me to PLEASE STOP writing articles for TopIcs. One can always hope. So, until next time . . . . (if there is a next time). . . . live long and prosper.

It's Only Fairware, But I Like It

by Steve Mehr, UG Member

Well, well, well, what do you know. Finally got Ray Kazmer to a L.A. meeting! What a joy it is to see Ray "perform" for us all. His presence has the same flavor as his programming. If you missed this meeting, you missed quite a lot!

At the present time we find Ray investing his programming efforts keeping the Fairware authors protected and duly represented. We currently find in Tex-Comp's catalog a section listed as "Demo_Disks". On these disks are the programming efforts from some of the finest Fairware authors in the 4/A community, including Ken Gilliland and Ray Kazmer. (Spoken with a French accent... "But of course!" Sorry Ray, G before K!) While the Fairware authors (that we know of) were not contacted by Tex-Comp for inclusion in their catalog, another source of distribution is created and hopefully will benefit the authors involved. So much for that. To help this cause, Ray has devised a unique program loading environment for each of these Demo Disks. Once in place on the disks, anyone purchasing them from Tex-Comp will unquestionably be advised of the Fairware status of the program, along with the author's name, address, and other pertinent information, all in Ray's recognizable programming flair. Ray's idea was in part due to the cryptic Fairware messages sometimes found (or not found at all) when running a Fairware program or reading its documentation. This way, purchasers will know that the cost paid to Tex-Comp goes directly to Tex-Comp and the author gets zippo. Bravo, Ray! Truly for the benefit of the community!

P.S. Remember that you can probably obtain most of the programs in question directly from your User's Group library, usually for a few dollars less, and it's possible that some of the funds may go directly back to the author as many User's Groups hold fund raisers to help support Fairware authors. [Ed. Note: LA 99ers User Group has a VERY extensive fairware library]

A KRACKER TALE

by Earl Raguse, LA 99ers

I have had a Gram Kracker for well over two flawless years now, but a couple of weeks ago it would not let the TI Title screen come up, and I couldn't get to the Kracker menu. I have grown so accustomed to that Kracker convenience, that when it failed to work, I was completely lost. Not only that, but with all the rumors I have heard and the articles I have read in the newsletters, running Craig Miller down, I feared the worst. Also since my GrandRam which I ordered back in September 87 hadn't arrived yet (but that's another story), I thought I would have to return to the computing Dark Ages .

I had also heard that when one called MG one only got to talk to an answering machine and that messages often went unanswered. The latter I had some experience with, because when my Kracker needed a battery about 6 months ago, after almost two years service, and I couldn't find one in the stores, I called MG to find out where to buy one. I did not get an answering machine, I got Craig himself before the first ring finished. He admitted that the batteries seemed hard to find, but since they had a supply of them he offered to send me one. Now that didn't sound like the ogre that some would have you believe that MG has become.

The battery arrived in the next day's mail, along with a no charge set of resistors, capacitors and diodes with full instructions for for making a minor mod to the board to reduce the periodic loss of Gram zero. I never had that problem, but I did frequently lose Grams one and two. He asked that I return $2.5@ for the battery. The least price for which I have seen similar batteries was $3.25!

To get on with my Kracker tale, I called MG at

714/599-1431, (the number in my Kracker manual) and again Craig answered before the first ring was over. He assured me that they would still fix a Gram Kracker, and moreover there would not be a charge for the repair, but he did think it only fair if I included a check for $5.00 to cover shipping and handling. Since I live only about 40 miles away, I arranged to personally drop it off; Craig gave me explicit directions how to find his house. When I left it off with Craig about Tuesday noon, he said they would get right on it and get it back to me as soon as possible. I had the distinct feeling that I certainly would get it back within two weeks at most.

On Thursday morning, I heard my door bell ring once. I went to the door and found no one, but I noticed a package lying behind my porch column. Yep, that's right, it looked just like my Kracker box with a UPS sticker, and it was, I could scarsely believe it. How's that for service?

All agog, I opened the box to find a note from Craig saying that they could not make my unit fail, also attached was a printed test result showing all the exhaustive testing that had been done. Bewildered and thinking I had had a bad dream, I ran to my computer and plugged the Kracker in, turned on the computer and there was the same problem!

I didn't know quite what to do, so I let it stew for a while so I could talk to others about it, but no one seemed to be of help. On Monday morning I again called Craig and again got an immediate answer. He was baffled. Finally it was arranged for me to return with all the computers which the Kracker had failed to work in (I have four, two black and two beige). Craig met me at the door and he plugged my Kracker into his computer and presto!! it worked just the way it should. He said that he used a beige console Version 2? as a worst case, thinking that if the Kracker worked in that it would work in anything.

Craig then hooked up one of my computers and plugged in the Kracker, turned on the computer and (I held my breath) NOTHING. The same for all the other computers. To make a longer story shorter, Craig was never able to determine what the problem was, with the limited trouble shooting equipment at his house, except that it was in my lower card. He replaced the card with one of his: then he went through an exhaustive testing routine to insure that he did not send me home with a faulty unit. All told Craig spent at least three hours and replaced the main card making a very very satisfied customer AT NO CHARGE!

Craig told me that they have always repaired anything that they sold and he fully intended to continue that policy. I hope never to hear or read again any further derogatory comments about MG because they try to protect their software or they have stopped manufacturing the Kracker or for working on IBM related products. I can't see how I could have been more fairly treated by any reputable manufacturer, of which MG is one of the most.

MICROdex 99, IN SEARCH OF...

by Steve Mehr, UG Member

MICROdex99 made its debut last month during the April meeting. Author and out-of-state member Bill Gaskill has produced a very useful and much needed indexing system for all TI related information. Similar in concept to John Owen's United 99/4 Data Base project, (herein referred to as U99/4DB) concept is the only thing the two projects share. I want to mention that John's project is very important to the TI community and needs your support. (See TopIcs Feb. 88 Vol. 7 No. 2 Pg. 12 for my review of his demo.) Similarly, Elton Schooling has been working on an index of his own and it appears his will be published in future issues of MICROpendium. So what does Bill's MICROdex99 system have to offer? Just let me tell you...

MICROdex99 is a stand alone program. By that I mean the program is self contained. The package consists of a system disk, and four library disks, labeled A-E, F-M, N-S, and T-Z. To access the data base, simply boot the system disk. No need to have (or understand) FR-BASE. (NO, NO, NO, I'm not knocking FR-BASE, but let's face it, there are those among us who feel intimidated by it's awesome powers.) Once MICROdex99 has been loaded, you may perform one or two level searches from A to Z for any information including Subject, Source, Type, Date, or Page. One thing to remember for those familiar with U99/4DB... MICROdex99 does not contain any other information about the item being searched. MICROdex99 is simply (and the key word here is "simply") "a published information referencing system." It simply tells you where to find that bit of information you read "somewhere." With UN99DB, you have to index 325 records before you can start searching the data base. Yes, the searches are fast, no doubt about it. And the searches in MICROdex99 are slow, again, no doubt about it. But, an item can be found much faster with MICROdex99 due to the lack of indexing at the start.

Currently, Bill Gaskill has unselfishly named MICROdex99 as a fund raiser for the L.A. 99er's Computer Group. (The key words here are "fund raiser".) For every 5 disk system sold, Bill is generously donating half the proceeds to our Treasury. The price for the complete 5 disk system is only $10.00! With the current library at around 3000 entries, the first update should be well over 4000, that's a steal. At the present time, MICROdex99 is being disributed only to members in good standing of the L.A. 99er's Computer Group, (another benefit of

membership) and I have been named by Bill to be the authorized distributor. Copies will be available from me at the meetings, or for those of you unable to attend, you may obtain the package from me by mail. For mail orders inside the USA, please send $13.00 to cover the MICROdex99 system, mailer, and postage. For orders outside the USA, please send $15.00 in US funds. Please make checks payable to Steve Mehr and send orders to:

Steve Mehr
633 Hollyburne Lane
Thousand Oaks, Ca. 91360
ATTN: MICROdex99

Library updates will be announced through this newsletter and will be handled by Bill Gaskill directly.

P.S. MICROdex99 system disk V1.1 is now available! Many new features have been added including two level searches, and the ability to search all drives for your library disks! Cut disk swapping on two drive systems in half, three drive systems by 2/3 and totally eliminate disk swapping on four drive systems! For purchasers of V1.0, don't forget to bring your system disk to the May meeting to receive your update! For mail orders, mail to the above address, "ATTN: MICROdex99 V1.1 Update". Please include $1.00 to cover mailer and postage. Thank you.

LI TOPICS

by Howie Rosenberg

A few Daffynitions:

BETA TESTER....A friend of the author who wants a free copy of the new program.

BETA TESTING.....The process of running a new program once to see if it does anything.

USER FRIENDLY....The documentation (if any) and menus(if any) are not written in Hindustani or Greek.

USER SUPPORT....The process of users supplying enough cash to support a programmer. "They OWE me support 'cause I support them". (An interesting use of two meanings of support).

DATA BASE PROGRAM.....II talk for another mailing list.

BUG FREE.....Does not lock up the console most of the time.

WELL DOCUMENTED.....Comes with either a piece of paper with writing on it or a disk file containing some text.

STATE OF THE ART....."The best I can do"

THE LATEST REVISION.....If you wait until tomorrow you can get a later one. Wait long enough and you may get one that works. In any case you can get updates sometimes even at a cheaper price than you paid for the original.

If you are a software producer and any of these definitions look good to you, please don't try to sell me anything you do.

Computer Chess and Chess on the 99/4

My use of 4 rather than 4A is quite intentional as the Video Chess module was the first module I bought shortly after I bought my first TI-99/4(without the A). For some time, I had been aware of work done in the area of developing chess playing programs as a subset of studies, largely government funded, for development of algorithms which were inductive in nature. Government interest was not, of course, in chess playing programs, but rather in such areas as language translation and such. Ever since the development of computers, programmers had tried to develop algorithms which "performed" inductive type processes. Development if Algebraic or Geometric theorems was another area which was pursued. In any case, Shannon (of information theory fame), early on came up with an algorithmic approach to chess programming which, from the outside looked much like the machine actually was thinking but actually was "brute force". That same approach is inherent in most every chess playing program to date(with one brand new exception which I shall mention later).

The sequence of a chess game can be described by a diagram which looks much like a tree. At the top level are all 20 moves which are possible for whites first move(16 pawn moves and 4 knight moves. Each of these moves has 20 replies for black. At each new point on the tree, a series of opponents moves branches out. After a very few moves the total number of possibilities is enormous, defying complete analysis and calculation by any computer including the largest. Shannons approach(and all who followed) was to calculate a previously defined function at each node(point on the tree) being analyzed. The function assigned values for such factors as material, mobility, control of the center in short all of the positional factors expounded by chess masters. The analysis would chew it's way down the tree as far as it could go in the time allocated for the move by the conditions set up in the program. The best score would be the move made at the end of the time period. Shortcuts were taken in the evaluation. For example,

early on the "Alpha Beta" algorithm was used(it still is today!). Simply put, if at one node, a score less than computed at another node for ones opponent is reached, no further branches at that node need be examined as it is assumed best play by the opponent(the usual "human" strategy in chess playing). Until quite recently all chess playing programs were quite similar in structure. Brute force chewing through the move tree with the faster machine examining more possibilities in the available time. Memory available became a large factor as opening "book' moves were stored to save time in the opening stages of a game.

Video Chess for the TI-99/4 made it's appearance early on. It was the first module I owned. There are many "sophisticated features in Video Chess. The simultaneous game mode is a great demonstration. If it were not for the inability to save and load to disk(most early modules assumed cassette based systems), the module would have been invaluable, early on, for chess players as a recording/retrieval device for games. As a learning tool it is great. Unfortunately it is quite poor as a chess player. The use of GPL(some assembly was used in Video Chess, I believe) and memory limitations make it fall far short of contemporary chess programs(machines). As many are aware, Gram Kracker(or substitute) owners can now modify Video Chess to make use of disk files thanks to the efforts of Mike Dodd thus making Video Chess more useful to Chess players as an archiving and game storage device. A similar capability is part of a new program announced by Asgard which does not require a Gram Kracker.

Video Chess has a maximum time per move of 2 minutes which according to the program results in a player of intermediate level. I never quite understood the fact that nowhere near the time available is used by Video Chess. It's almost as if TI said at some point in development "that's good enough wrap and ship". The available time it does not utilize could well be used in finding somewhat better moves than the program does. How poor the program performs was clear to me when I matched Video Chess at 2 minutes per move against an AT clone at 5 seconds per move using ChessMaster a rather good modern Chess program. Not only did CM destroy Video Chess in a very few moves, a repeat performance at Queen odds (in addition to the time odds)also led to the defeat of Video Chess! It is not in the area of superior play that Video Chess is of value but rather as stated above as a learning tool and record keeper.

Unfortunately the only other Chess program available for the TI is a recent arrival which is a translation of the original version of SARGON, originally programmed for the Z80 microprocessor. It must have been a formidable task to accomplish translating Z80 code. The end result unfortunately is extremely slow. So slow that play on any level above one is too time consuming to be worthwhile. Some experimentation in simple game positions lead me to believe that the Alpha Beta algorithm was not properly incorporated in this program. Thus ALL variations would need be calculated which contributes to the lack of speed. Accompanying Sargon is another file with the same graphics which is quite nice. A terminal program for chess players to allow playing via modem with the board on the screen in front of you. Very nice!

As of late changes are taking place in the basic design of chess playing programs. Rather than the brute force approach first presented by Shannon so many years back, pattern recognition is used as the basis of analysis. The prime force behind this approach is Hans Berliner who is the designer of the present worlds champion chess playing program. I remember Mr Berliner from about 25 years back when he was the perpetual winner each year of the US Postal Chess Championship as well as the worlds Postal Chess Champion on more than one occasion. I wonder if computer chess will make competitive postal chess impractical. Certainly someone with access to some of the present programs/machines which play at the master level will achieve results far superior to his human capabilities!

REMEMBER NEXT MEETING - Wednesday May 25, Torrance Public Library, 7 PM

CLUB OFFICERS			
President:		Librarian:	
Tom Freeman	(213) 454-1943	Fred Moore	(213) 670-4293
Vice President:		Library Assistants:	
Terrie Masters	(213) 271-6930	Alan Whiteman	(213) 379-8031
Secretary:		Chick De Marti	(213) 532-8499
Doug Moore	(213) 451-1069	TopIcs Editors:	
Treasurer:		Tom Freeman	(213) 454-1943
Alan Whiteman	(213) 379-8031	Terrie Masters	(213) 271-6930
Membership Chairman:		Equipment:	
Ed May	(213) 644-6241	Steve Chalcraft	(213) 929-3213
Sales Chairman:		Hospitality Chairman:	
Gail Fair	(213) 326-6660	Myron Harms	(213) 675-3959

Membership in the LA 99ers, including subscription to TopIcs is $20.00 per year

LA 99er BBS PHONES:

213-755-7239 (Danny Nelson, Sysop) and 213-864-2488 (Steve Chalcraft, Sysop)

LA99 LIBRARY CORNER

Copies Of all program disks will be made available to the members at the regular meetings. If you plan to obtain any disks from the library at the meeting it is best to phone or write the LIBRARIAN **in advance** to be sure they will be on hand. I will put your name on them.

ØØØØ **LA99ers LIBRARY CATALOG** APRIL 88 : SPECIAL PRICE $1.ØØ

NEW ADDS FOR MAY LA99 LIBRARY

The Library Committee wish to give thanks to those who donated disk to our Library this month : Danny Nelson, Terry Masters, Tom Freeman, Steve Mehr.

2429 **GRAPHIC LABELER** V3.Ø Fairware by Steve McWatty RR#1 Kinburn,ONT KOA 2HØ CANADA. To use Graphic use Dave Rose characters set. Prints 5 rows label with 1st line double emphasied, auto center and with graphic. A much improved easiser to use program. With many graphic on disk. SSSD(114).

2604 **DM1ØØØ 4*Ø** Fairware by Ottawa 99/4 U.G. P.O. BOX 2144 Station D. Ottawa,ONT, K1P5W3. Up date version 4*Ø. OK for TI, CorComp, Myrac, Geneve controller. To use with replace MGR1 and MGR2 plus MGR3. Use doc of 3*5 (SSSD)73

2645 **FUNLWRITER V4.Ø** DEC.87 Issue. Fairware by Tony & Will McGovern 215 Grinsell St. Kotara, NSW 2288 Australia. This Version Greatly improves TI-WRITER, EDITOR ASSEMBLY , DISK MANGER 1ØØØ (3.5) & UTILITI. all interconected with each other. Must read DOCS to get the complete infomation. 2(SSSD)698

2659 **GRAM KRACKER UTILITY III** By Tom Lidstone, Bernolillo User Group. This Gram Kracker Utility disk will allow you to personalize the screen with your own art work (using TI-Artist). Store Instances in the GK along with TI-Writer, DM1ØØØ, X/B, E/A #5, E/A #3 plus other you may choose. The main menu can have 1Ø menu items. There are other items that you can do read the docs. SSSD(181).

2839 **ENCODE-IT** V1.Ø Fairware by Peter Hoddie and Jim Lohmeyer 12 Paul Revere Rd. Lexington, MA Ø2173 : A general purpose encryption utility program designed to allow encryption of a file or an entire disk. Keep your program secret. SSSD(45)

7Ø58 **WAGNER'S** Fairware by By Ken Gilliland 543 Riverside Dr. #15 Glendale, CA 912Ø4. Four songs by Wagner's DAS RHEINGOLD (prelude to act I), DIE WALKURE (prelude to act III), SIEGFRIELD (predude to act I), GOTETTERDAMMERUG (predude to act III). All with story about the song and with great graphic well done as all of Ken programs are. 1SSSD(281),1SSSD(214)

7Ø59 **MUSIC 48** Fairware by Jim Pederson (Tigerclub) 156 Columbus, OH 43213. 22 musicial songs with words and great graphic. ANYTIME, CANDY, COLUNBUS, DARK, DEVILWOMAN, DIXIE, DON'T WORRY, EL PASO, FRANKIE AND JOHNNIE, GREENSLEEVE, HOME DREAM, KOLO, NELLIE, ROSES, SAN ANGELO, SHENAANDOAH, TARNISHED, TILL WALSH, WAY, WEDDING, WILDROOD, WISHING. AUTO PLAYS ALL OR ANY SONGS. SSSD(358).

LIBRARY ADDS FOR MAY, CONTINUED

8Ø99 **STARFLEET** By Stephen Barackman. Color pictures of the USS ENTREPRISE NCC 17Ø1, NCC 17Ø1A and 17Ø1D of the United Federation Of Planets. Use Display Master to view or edit. Load file >DSK1.STRFLTE. SSSD(36Ø)

81ØØ **STARFLEET 1** By Stephen Barackman. Color pictures technical drawing (top secert) of other USS Federation of Planets Starfleet. The USS RELIANT NCC 1864, the USS VENTURE NCC 1874 and the USS STINGRAY NCC 1852. Use Display Master to view or edit. Load file >DSK1.STRFLT1. SSSD(359)

81Ø1 **STARFLEET 2** By Stephen Barackman. Color pictures technical drawing (top serect) of star ships in the Klingon Empire. The ISS DEVISOR the ISS BLUEFIRE and the ISS KAHLESS use Display Master to view or edit. Load file >DSK1.STRFLT2. SSSD(359)

8102 **INSTANCES #7** 24 pictures in DIS/VAR 8Ø format. BIRD, BOO, CARE, CLOWN, DOCTOR, DONALD, DUMBO, FOOTBALL, GOOSE, H-HEAD, HAPPY, HORSE2, HORSE, LAMB, LOVETHY, MAN2, MAN, MARY, MICKEY, PLUTO, POSTMAN, PUMPKIN, SANTA, SCACROW. SSSD(357.

8103 **FONTS #1** 11 Style printing in DIS/VAR 8Ø format. 3D, ABC, BSBOLD, FAREAST, FBLOCK, GOTHIC, IBM, LEDGER, LED, OFFBEAT. SSSD(352)

8104 **FONTS #2** 14 Style printing in DIS/VAR 8Ø format. LOWEROE, OLDENG3, OLDENG, PLAIN, PLANE, PLAYBIL, RODEO2, ROMAN, ROUND, SCRIPT, SCRIPT5, SHELBY, SLANT, TECH2. SSSD(338)

9Ø79 **GAMES #59** Five games from Germany great color, action, and graphic automatic loan X/B. need to understand German to play the games. : ARTILLERIE, CAVEMAN, DIE FRAGGLES, HERKULES, NOAH 2Ø99. SSSD(233)

9Ø8Ø **MONOPLY** Fairware by Steve Karasek 855 Diversey, ST.Louis, MO 63126. A computer game just like the board game with docs. Up to 1Ø players. The computer will do all the work roll the dice, move your player and figure the pay off. Good color and graphic. SSSD(1Ø6).

9Ø81 **CARFAX ABBEY** By David Vincent A 3D maze type adventure game. You are in a Abbey and need to locate and destroy Count Dracula. Advance layout for the expects as well as the beginner. great graphic docs and help on disk. SSSD(236).

LIBRARIAN
FRED MOORE
7730 EMERSON AVE.
LOS ANGELES,CA 9ØØ45
213-67Ø-4292

BUG SPRAY - a quickie from Bill Gaskill

Bill writes to tell us there is a small error in his EZ-KEYS review from April TopIcs. The custom character set provided does NOT have any letters out of line. The problem was that the program Bill was RUNning redefined a couple of them! Apologies to the author Harry Wilhelm.

MARKETPLACE

(the marketplace is a fund raiser for the club, that is, the "profit" goes to maintain the quality of this News- letter. In general the price listed splits the difference between cost and retail. Please help your Club.)

SPECIAL - SUPER EXTENDED BASIC by Triton - code by MG & friends 50.00
plus P&H 2.50

SPECIAL #2 - WE HAVE THREE HORIZON RAMDISK, DIFF.CONFIGS - CALL FOR PRICES

MILLERS GRAPHICS

DISKASSEMBLER	18.50	ORPHAN CHRONICLES (PRICELESS)	9.95
ADVANCED DIAGNOSTICS	18.50	NIGHT MISSION	18.50
GK UTILITY I	10.00	SMART PROGRAMMING FOR SPRITES	6.25

GENIAL COMPUTERWARE

XBasher (MIKE DODD)	9.00	XB:Bug (J.PETER HODDIE)	12.00
GRAM PACKER (JPH) "	9.00	REMIND ME! (JOHN JOHNSON)	12.00
PC TRANSFER (MD)	20.00	FONT PACK I (JPH)	9.00
GRAPHICS EXPANDER(JPH)	9.00		

RYTE-DATA

GPL SETS (INCLUDING ASSEMBLER AND LINKER, 4 DISKS	50.00	COMMAND DOS (MONTY SCHMIDT)	20.00
		TECHNICAL DRIVE(BOOK BY ")	15.00
BASIC COMPILER	15.00	SUPER CLOCK SUPPORT	13.50

BYTEMASTER (R. MITCHELL)

MG EXPLORER (UNPROTECTED)	20.00	STRINGMASTER	16.00

KRACKER FACTS (MIKE DODD, ED.)	5.00	UTILITIES DISK/DOCS (T FREEMAN)	8.00
ORPHAN SURVIVAL HNDBK(ALBRIGHT)	15.00	JOYPAINT	30.00
JOYPAINT PAL	7.50	CERTIFICATE 99	20.00
FONT WRITER II (JPH)	19.00	PRE-SCAN IT! (J.PETER HODDIE)	10.00
TPA FONTS DISKS 1 OR 2	9.50	PRINTER'S APPRENTICE (M.McCANN)	19.00
TPA AND FONTS DISK 1(SET)	26.50	TPA TOOLBOX	19.00

MYARC PRODUCTS, INCLUDING GENEVE - check for discount prices

INSCEBOT

TI-ARTIST	15.00	DISPLAY MASTER	12.00
ARTIST EXTRAS	6.00		

MEGATRONICS

EXTENDED BASIC II PLUS	72.50	INTERN (BOOK ON GPL)	16.50
128K GRAM CARD	227.50		

HARDWARE & SUPPLIES

TEAC 55BV DSDD DRIVES	90.00	DISKETTES DSDD	.50
TECHNICAL AND BUSINESS BOOKS	5.00		

REPRINTS

HANDY REFERENCE GUIDE	2.50	LOGO DIGEST	2.50
BEST OF NEWSLETTERS W/DISK	5.00	FORTH NOTES VOL 1-6 (2.50 EA)	10.00
BEGINNER'S FORTH NOTEBOOK	2.50	ASSEMBLY NOTES VOL 1	2.50

BACK ISSUES

SMART PROGRAMMER MISC. SET OF 8	4.00	MICROPENDIUM	1.25

(please send your order to the CLUB address, not the Librarian, and add $1.00 per item for P & H ($2.50 for Super XB). CA residents add 6.5% tax).